REMARKS

Claims 1, 2, 3, 8, and 29-32 have been rejected under 35 U.S.C. § 102(e) as being anticipated by Brown et al. (U.S. Patent No. 6,604,075; hereinafter "Brown-2003"), which references Brown et al. (U.S. Appl. No. 09/168,405, hereinafter "Brown-2001"). Claims 9, 10, 19, 20, and 37-39 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Brown-2001 in view of Kantrowitz et al. (U.S. Pat. 6,618,697). Claims 11-14, 17, 18, 21-24, 27, 28, and 33-36 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Brown-2001 in view of Kantrowitz et al. and taken in further view of Zadrozny et al. (U.S. Pat. 5,937,385). Claims 15, 16, 25, and 26 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Brown-2001 in view of Kantrowitz et al. and taken in further view of Zadrozny et al. and Applicant's admitted prior art. Reconsideration of these claims is respectfully requested.

Brown-2003 discloses a system that provides an improved voice dialog interface for use in web-based applications implemented over the Internet or other computer network. Col. 2, ll. 32-34. The interface in an illustrative embodiment includes a web page interpreter for receiving information relating to one or more web pages. Col 2, ll. 39-41. The web page interpreter generates a rendering of at least a portion of the information for presentation to a user in an audibly-perceptible format. Col. 2, ll. 41-44. The system receives HTML information from the Internet or other computer network in an HTML interpreter which processes the HTML information to generate a rendering, i.e., an audibly-perceptible output of the corresponding HTML information for delivery to a user. Col. 7, ll. 31-35. The HTML information is also delivered to a grammar compiler which processes the information to generate a syntax and a set of lexical semantics. Col. 7, ll. 36-39. A grammar generation process can receive as input parsed HTML and generate GSL (Grammar Specification Language) therefrom. Col. 7, ll. 53-55. The grammar compiler may be configured to take this GSL as input and create an optimized finite-state network for a speech recognizer. Col. 7, ll. 55-57. More particularly, the GSL may be used, e.g., to program the grammar compiler with an expanded set of phrases so as to allow a user to speak partial phrases taken from a hyperlink title. Col. 7, ll. 57-60. Speech received from a user is processed in an automatic speech recognizer (ASR) utilizing the syntax generated by the grammar compiler. Col. 8, ll. 1-4. The output of the ASR is applied to a natural language interpreter which utilizes the lexical semantics generated by the grammar compiler. Col. 8, ll. 4-6. When a user speaks a client-side command, such as "speak faster" or "speak louder," the command is executed immediately and the presentation continues. Col. 10, ll. 29-31. When a navigation command associated with a hyperlink is spoken, control is transferred to the corresponding new web page, dialog turn, and presentation and speech grammar context. Col. 10, ll. 31-34. The process can then continue on to a new dialog state. Col. 10, l. 35. Dialog control may be handled by representing a given dialog turn in a single web page. Col. 3, ll. 5-7. In this case, a finite-state dialog controller may be implemented as a sequence of web pages each representing a dialog turn. Col. 3, ll. 7-9.

Brown-2001 discloses apparatus and methods for implementing Interactive Voice Response (IVR) applications over the Internet or other computer network. Para. [004]. The voice processor 114 in IVR platform 102 takes output from the HTML parser 112 and further analyzes the corresponding retrieved HTML web page to identify structure such as, for example, section headings, tables, frames, and forms. Para. [0017]. The HTML parser 112 parses HTML in retrieved web pages for the purposes of facilitating production of speech output and generation of grammar. Para. [0027]. The voice processor 114 in conjunction with TTS synthesizer 116 then generates a corresponding verbal description of the page. Para. [0017]. The parsed HTML is analyzed in grammar generator 120 for section titles, hyperlinks and other indicators that are to be converted to speech. Para. [0021]. Grammar generator 120 then constructs a subgrammar for each indicator by generating all possible ways of speaking subsets of the indicator. Para. [0021] All other voice commands are then combined with the subgrammar and a complete grammar is compiled into an optimized finite-state network. Para. [0021]. This network is loaded into the speech recognizer 122 to constrain the possible sequences of words that can be recognized. Para. [0021]. This process provides a tightly constrained grammar with low perplexity that allows all possible word deletions to be spoken, thereby giving the user freedom to speak only the smallest set of words necessary, e.g., to address a given hyperlink. Para. [0061]. The process can also create many redundancies in the resulting GSL description, because leading and trailing words are reused in many subsets. Para. [0061]. This inefficiency may be reduced in four stages of code optimization described in Brown-2001. Para. [0063].

Kantrowitz et al. disclose a rule-based method for detecting and correcting spelling and grammar errors. Col. 1, ll. 53-55. The method comprises the steps of storing a plurality of spelling rules defined as regular expressions for matching a potentially illegal n-gram which may comprise less than all letters in the word and for replacing an illegal n-gram with a legal n-gram to return a corrected word. Col. 2, ll. 6-11. Furthermore, the method comprises storing a plurality of spelling and grammar rules defined as regular expressions given the context of one or more adjacent words. Col. 2, ll. 18-20.

Claim 1 is patentable by calling for a method of developing an interactive system, including "generating a dialogue state machine, including a number of procedures with variables, on the basis of said application data, said state machine including slots for each operation and each input parameter." Support for the amended limitation can be found on page 5 of the specification of the present application, wherein it states that the dialogue state machine is, for example, a finite state machine, written in the ITU's Specification Description Language, including a number of procedures with variables. In contrast, Brown-2003 does not disclose, teach, or suggest a process for generating a dialogue state machine, including a number of procedures with variables, on the basis of application data, the state machine including slots for each operation and each input parameter. Rather, the system receives HTML information from the Internet or other computer network in an HTML interpreter which processes the HTML information to generate a rendering, i.e., an audibly-perceptible output of the corresponding HTML information for delivery to a user. Col. 7, ll. 31-35. The web page interpreter may make use of certain pre-specified voice-related tags, e.g., HTML extensions. Col. 2, ll. 44-46. Brown-2003 discloses using a number of extensions to conventional HTML. Col. 3, ll. 60-62. The HTML extensions may be embedded in the scope of an HTML anchor. Col. 4, ll. 1-2. However, there is no element of HTML that relates to "a dialogue state machine, including a number of procedures with variables, . . . said state machine including slots for each operation and each input parameter." Furthermore, Claim 1 recites "generating grammar . . . said grammar including slots for each operation and input parameters to return data of said parameter types to said state machines." There is no element of HTML that defines the types of parameters of the kind described in Applicants' specification and claims.

Brown-2003 discloses that although the illustrative embodiment utilizes HTML, the invention can be implemented in conjunction with other languages, e.g., Phone Markup Language (PML), Voice eXtensible Markup Language (VoiceXML), Speech Markup Language (SpeechML), Talk Markup Language (TML), etc. Col. 3, ll. 62-67. However, none of these languages are abstract descriptions of the operations and input and return parameters of a dialog system. Rather, these languages are detailed, low level application programming languages. Therefore, Claim 1 is patentable over Brown-2003.

Claims 2, 3, 6, 8-18, 33-35, and 39 depend from Claim 1 and are patentable for at least the same reasons as Claim 1 and by reason of the additional limitations called for therein. Claims 29 and 30 rely on the patentability of Claim 1 and are patentable for the same reasons as Claim 1 and by reason of the additional limitations called for therein.

Claim 19 is patentable by calling for a grammatical inference method for developing grammar, including "creating additional rules representative of repeated phrases . . . wherein said rules define slots to represent data on which an interactive system executes operations and include symbols representing at least a phrase or term." Neither Brown-2001 nor Kantrowitz et al. teach or suggest "creating additional rules representative of repeated phrases . . . wherein said rules define slots to represent data on which an interactive system executes operations and include symbols representing." In contrast, Brown-2001 discloses that grammar generator 120 constructs a subgrammar for each indicator by generating all possible ways of speaking subsets of the indicator. Para. [0021]. All other voice commands are then combined with the subgrammar and a complete grammar is compiled into an optimized finite-state network. Para. [0021]. The process creates many redundancies in the resulting GSL description. Para. [0061]. This inefficiency may be reduced in four stages of code optimization. Para. [0063]. However, Brown-2001 does not relate to grammatical inference. Brown-2001 discloses only grammar optimization. That is, the merged finite state networks describe the same language as the pre-merged finite state networks. Furthermore, the slots as claimed in Claim 19 represent data on which an interactive system executes operations. Brown-2001 discloses that HTML is analyzed in grammar generator 120 for section titles, hyperlinks and other indicators that are to be converted to speech but does not disclose slots representing data on which an interactive system executes operations and include symbols representing at least a phrase or term. Further yet, the method of Claim 19 includes creating additional rules representative of repeated phrases, e.g., sequences of words that appear in more than one context. As the Examiner points out, Brown-2001 does not disclose creating additional rules representative of repeated phrases.

Contrary to the Examiner's assertion, Kantrowitz et al. do not teach or suggest "creating additional rules representative of repeated phrases . . . wherein said rules define slots to represent data on which an interactive system executes operations and include symbols representing at least a phrase or term." Rather, Kantrowitz et al. disclose storing a plurality of spelling and grammar rules defined as regular expressions given the context of one or more adjacent words. Col. 2, ll. 18-20. Kantrowitz et al. disclose an "of of" rule, which corrects a common example of repeated words. Col. 4, ll. 20 and 31-32. The motivation in the present invention for creating additional rules representative of repeated phrases is to identify interchangeable phrases, and thus extend the language model represented by the grammar by creating additional rules in the process of grammatical inference. The method disclosed in Kantrowitz et al. for identifying repeated words in a spelling and grammar check process is unrelated to grammatical inference. Furthermore, Kantrowitz et al. refer to repeated words, i.e., words that are repeated one after another, not "rules representative of repeated phrases . . . wherein said rules define slots to represent data on which an interactive system executes operations and include symbols representing at least a phrase or term," as recited in Applicant's Claim 19.

Claims 20-28 and 36-38 depend from Claim 19 and are patentable for at least the same reasons as Claim 19 and by reason of the additional limitations called for therein. Claims 37 and 38 rely on the patentability of Claim 19 and are patentable for the same reasons as Claim 19 and by reason of the additional limitations called for therein.

Claim 31 is patentable for reasons similar to those discussed above with respect to Claim 1 by calling for a system for developing an interactive system, including "means for generating a dialogue state machine, including a number of procedures with variables, on the basis of said application data, said state machine including slots for each operation and each input parameter."

Claim 32 is patentable for reasons similar to those discussed above with respect to Claim 1 by calling for a development tool for an interactive system, including "code for generating a dialogue state machine, including a number of procedures with variables, on the basis of said application data, said state machine including slots for each operation and each input parameter, said slots defining data on which said interactive system executes the operations."

In view of the foregoing, it is respectfully submitted that the claims of record are allowable and that the application should be passed to issue. Should the Examiner believe that the application is not in a condition for allowance and that a telephone interview would help further prosecution of this case, the Examiner is requested to contact the undersigned at the phone number below.

Respectfully submitted,

DORSEY & WHITNEY LLP
Customer Number 75149

Date: July 1, 2008

By:
Nathan J. Witzany
Reg. No. 60,948
Telephone: (612) 492-6862

4839-3472-3073\1